ATTACHMENT- 4

4 a) Alternator Subsystem

CSSL-IV Program Listing

```
PROGRAM ADEM
" Automotive Electrical System Model "
" Technical Consultants: "
" Ronald Krefta-(765)451-3782: synchcronous generator with 3-phase "
" delta connected stator winding and dc-excited rotor winding. "
" TV Sriram-(765)451-3821: 3-phase diode rectifier bridge circuit."
" Mike Bradfield-(765)649-3049: rotor field current PWM controller. "
" Steve Cochran-(317)579-3730: lithium polymer(LiPo) battery "
" Technical Coordinator: Dr. John McBain-(765)451-3739 "
" Program Developer: Dr. Yilmaz Sahinkaya-(650)574-0254 "
" Model Creation Date: July 26, 2000 "
                    " Units : Metric "
                    " System Parameters "
" General Parameters"
" TFIN = Simulation Time (sec) "
CONSTANT TFIN    = 100.0E-3
" Engine Speed Command Model Parameters "
CONSTANT TSENG   =     0.0            $" Engine Starting Time (sec) "
CONSTANT TRIDLE  =     5.0            $" Idle Rise Time (sec) "
CONSTANT NVIDLE  =   150.0            $" Idle Rise Rate (rpm/sec) "
CONSTANT TIDLE   =     8.0            $" Idling Time (sec) "
CONSTANT NIDLE   =   700.0            $" Idling Speed (sec) "
CONSTANT TRENG   =    10.0            $" Engine Rise Time (sec) "
CONSTANT NVRENG  =   175.0            $" Engine Speed Rate (rpm/sec) "
CONSTANT TCRUISE=     18.0            $" Engine Cruise Time (sec) "
CONSTANT NCRUISE= 2500.0              $" Engine Cruise Speed (rpm) "
CONSTANT TFENG   =    10.0            $" Engine Fall Time (sec) "
CONSTANT NVFENG  =   175.0            $" Engine Speed Fall Rate "
CONSTANT NEZ     =   640.0            $" Initial Engine Speed (rpm) "
CONSTANT MODCON  =     1.0            $" MODCON = 1.0, Gen Control is on"
                                      $" MODCON = 0.0, Gen Control is off"
CONSTANT MODTEST=      0.0            $" MODTEST = 0.0, Gen Test is off "
                                      $" MODTEST = 1.0, Gen Test is on  "
CONSTANT VGBTEST=     44.0            $" Gen Bus Test Voltage (Volts) "
CONSTANT IFGENZ  =     4.9            $" Gen Field Test Current (Amps) "
" 42 Volt Loads "
" Starter Motor Parameters "
" TSMON, TSMOFF = Starter Motor ON, OFF Times (Sec) "
CONSTANT TSMON   = 0.0, TSMOFF = 10.0E-3
CONSTANT KTSM    =     0.7            $" Torque Constant (Nm/Amps) "
CONSTANT ISMAMP  =     150.0          $" Motor Current (Amps) "
CONSTANT RSM     =     0.025          $" Armature Resistance (Ohms) "
" Power Steering (PES) Motor Parameters "
" TPSON, TPSOFF = Power Steering Motor ON, OFF Times (Sec) "
CONSTANT TPSMON = 20.0E-3, TPSMOFF = 50.0E-3
CONSTANT KTPSM   =   1.0              $" Torque Constant (Nm/Amp) "
CONSTANT IPSMAMP=    30.0             $" Motor Current (Amps) "
CONSTANT RPSM    =   0.25             $" Armature Resistance (Ohms) "
CONSTANT OMGPS   =   1.50             $" Steering Frquency (Rad/Sec) "
" Generator Parameters "
CONSTANT VGBREF  =    42.0            $" Generator Bus Reference(Volts)"
CONSTANT GRGEN   =     2.5            $" Generator Gear Ratio "
CONSTANT NPPGEN  =     6.0            $" Number of Pole Pairs "
" LST = Synchronous Inductance(Henry) vs Generator Speed (rpm) Table "
TABLE LST, 1, 12,...
     0.0, 1600.0, 1800.0, 2000.0, 2500.0, 3000.0,...
     3500.0, 4000.0, 5000.0, 6500.0, 8000.0, 10000.0,...
     1.77E-3, 1.77E-3, 1.846E-3, 1.924E-3, 2.135E-3, 2.275E-3,...
     2.37E-3, 2.43E-3, 2.49E-3, 2.52E-3, 2.53E-3, 2.53E-3
```

Figure 15. The CSSL-IV Program for the Delphi Alternator Math Model

```
" Stator Phase Resistances at 25 deg C (Ohms)"
CONSTANT RAGENO = 0.29, RBGENO = 0.29, RCGENO = 0.29
" ALPHAW  = Winding Resistance Thermal Coefficient /deg C Rise "
CONSTANT ALPHAW = 0.00394          $" Ohms Per Deg C Temp Rise "
" LMFT = Stator Phase-Rotor Field Mutual Inductance Table Function "
" First Independent Variable = Stator Phase Current(Amps)"
" Second Independent Variable = Rotor Field Current (Amps)"
" Dependent Variable = Mutual Field-Phase Inductance (Henry) "
TABLE LMFT, 2, 6, 12,...
      0.0, 0.1, 2.0, 4.9, 6.0, 7.0,...
      0.0, 1600.0, 1800.0, 2000.0, 2500.0, 3000.0, 3500.0,...
      4000.0, 5000.0, 6500.0, 8000.0, 10000.0,...
      0.0205, 0.0205, 0.0202, 0.0110, 0.0092, 0.0092,...
      0.0205, 0.0205, 0.0202, 0.0110, 0.0092, 0.0092,...
      0.0227, 0.0227, 0.0224, 0.0122, 0.0102, 0.0102,...
      0.0242, 0.0242, 0.0239, 0.0130, 0.0108, 0.0108,...
      0.0277, 0.0277, 0.0274, 0.0149, 0.0124, 0.0124,...
      0.0301, 0.0301, 0.0298, 0.0162, 0.0135, 0.0135,...
      0.0318, 0.0318, 0.0314, 0.0171, 0.0142, 0.0142,...
      0.0329, 0.0329, 0.0325, 0.0177, 0.0147, 0.0147,...
      0.0338, 0.0338, 0.0334, 0.0182, 0.0151, 0.0151,...
      0.0344, 0.0344, 0.0339, 0.0185, 0.0153, 0.0153,...
      0.0345, 0.0345, 0.0341, 0.0186, 0.0154, 0.0154,...
      0.0345, 0.0345, 0.0341, 0.0186, 0.0154, 0.0154

" LFGEN  = Field Self Inductance (Henry) "
" Note: LFGEN is a nonlinear function of IFGEN(Field Current(Amps))"
TABLE LFGENT, 1, 11,...
      0.0, 0.1, 0.5, 1.0, 2.0, 3.0, 4.0, 4.9, 5.0, 6.0, 7.0,...
      0.87, 0.87, 0.98, 1.0, 0.92, 0.76, 0.625, 0.54, 0.53, 0.46, 0.46
" RFGENO  = Field Resistance (Ohms) "
CONSTANT RFGENO = 2.055            $" Ohms at 25 deg C "
" TIGREF  = Generator Interior Reference Temp ( deg C) "
CONSTANT TIGREF = 25.0             $" deg C "
" LEC Regulator Parameters "
CONSTANT GPLEC  = 10.0, GILEC = 1.0
CONSTANT DREGMX = 0.328            $" Regulator Max Duty Cycle "
CONSTANT VLECPL =  6.0             $" Positive LEC Limit (Volt) "
CONSTANT VLECNL = -6.0             $" Negative LEC Limit (Volt) "
CONSTANT VGBCONZ=  0.0
" Initial Conditions on State Variables "
CONSTANT IAGENZ = 0.0, IBGENZ = 0.0, ICGENZ = 0.0
CONSTANT TAUDEC = 6.6E-6           $" Current Decay Time (Sec) "
CONSTANT TAUDIF = 10.0E-6          $" Dif. Model Time Constant(sec)"
" The 42 Volt Battery Load Dump Test Parameters "
" TLD42 = Load Dump Test Starting Time (Sec) "
CONSTANT TLD42 = 1000.0
" SWLD42 = Load Dump Switch "
" SWLD42 = 0.0, Load Dump is OFF "
" SWLD42 = 1.0, Load Dump is ON "
CONSTANT SWLD42 = 0.0
" Rectifier Parameters "
" Avalanche Diode Forward and Reverse Conduction Table "
" VD    = VDT(I)          (Volts) "
TABLE VDT, 1, 11,...
-100.0,-55.0,-35.0,-25.0,-15.0,-0.1, 0.0, 0.1, 35.0, 50.0, 100.0,...
-95.0,-95.0,-95.0,-94.0,-93.0,-92.0, 0.0, 0.85, 0.90, 0.95, 0.95
" Lithium Polymer Battery Parameters "
" Electric Analog Circuit Parameters "
" Battery Storage Capacitance Parameters "
```

```
CONSTANT VCMN    = 3.7              $" Minimum Cell Voltage (Volts) "
CONSTANT VCMX    = 4.7              $" Maximum Cell Voltage (Volts) "
" CSBT  = Battery Storage Capacitance (F) vs Current (Amps) "
TABLE CSBT, 1, 13,...
-200.0, -100.0, -75.0, -50.0, -25.0, -5.0, 0.0,...
 5.0, 25.0, 50.0, 75.0, 100.0, 200.0,...
 4.0E+5, 4.0E+5, 3.5E+5, 3.0E+5, 2.25E+5, 2.25E+5, 1.0E+5,...
 2.25E+5, 2.25E+5, 3.0E+5, 3.5E+5, 4.0E+5, 4.0E+5
" CSBLT = Limited Battery Storage Capacitance (F) vs Current (Amps) "
TABLE CSBLT, 1, 13,...
-200.0, -100.0, -75.0, -50.0, -25.0, -5.0, 0.0,...
5.0, 25.0, 50.0, 75.0, 100.0, 200.0,...
4.5E+4, 4.5E+4, 2.5E+4, 3.0E+4, 3.0E+4, 3.0E+4, 3.0E+4,...
3.0E+4, 3.0E+4, 3.0E+4, 3.0E+4, 3.0E+4, 3.0E+4
" VCON1 = Cell-1 Maximum Voltage (Volts) "
CONSTANT VCON1   = 4.2
" VCON2 = Cell-2 Maximum voltage (Volts) "
CONSTANT VCON2   = 4.2
" VCON3 = Cell-3 Maximum Voltage (Volts) "
CONSTANT VCON3   = 4.2
" VCON4 = Cell-4 Maximum Voltage (Volts) "
CONSTANT VCON4   = 4.2
" VCON5 = Cell-5 Maximum Voltage (Volts) "
CONSTANT VCON5   = 4.2
" VCON6 = Cell-6 Maximum Voltage (Volts) "
CONSTANT VCON6   = 4.2
" VCON7 = Cell-7 Maximum Voltage (Volts) "
CONSTANT VCON7   = 4.2
" VCON8 = Cell-8 Maximum Voltage (Volts) "
CONSTANT VCON8   = 4.2
" VCON9 = Cell-9 Maximum Voltage (Volts) "
CONSTANT VCON9   = 4.2
" VCON10= Cell-10 Maximum Voltage (Volts)"
CONSTANT VCON10  = 4.2
" RIB  = Battery Internal Resistance (Ohms) "
" RIB varies with TIB = Interior Battery Temperature "
" TIB  = Interior Battery Temperature (Deg C) "
" CFRBT= Temperature Correction Factor for TIB "
" Cell Resistance values at 25 Deg C (Ohms) "
CONSTANT RIC1I = 0.00250, RLC1I = 4.65E+3, RCON1I = 36.0   $" Cell-1 "
CONSTANT RIC2I = 0.00250, RLC2I = 4.65E+3, RCON2I = 36.0   $" Cell-2 "
CONSTANT RIC3I = 0.00250, RLC3I = 4.65E+3, RCON3I = 36.0   $" Cell-3 "
CONSTANT RIC4I = 0.00250, RLC4I = 4.65E+3, RCON4I = 36.0   $" Cell-4 "
CONSTANT RIC5I = 0.00250, RLC5I = 4.65E+3, RCON5I = 36.0   $" Cell-5 "
CONSTANT RIC6I = 0.00250, RLC6I = 4.65E+3, RCON6I = 36.0   $" Cell-6 "
CONSTANT RIC7I = 0.00250, RLC7I = 4.65E+3, RCON7I = 36.0   $" Cell-7 "
CONSTANT RIC8I = 0.00250, RLC8I = 4.65E+3, RCON8I = 36.0   $" Cell-8 "
CONSTANT RIC9I = 0.00250, RLC9I = 4.65E+3, RCON9I = 36.0   $" Cell-9 "
CONSTANT RIC10I= 0.00250, RLC10I= 4.65E+3, RCON10I= 36.0   $" Cell-10"
TABLE CFRBT, 1, 7,...
-45.0, -29.0, -18.0, 0.0, 25.0, 52.0, 75.0,...
 3.0, 3.0, 2.0, 1.2, 1.0, 1.0, 1.0
" Initial Values of State Variables "
CONSTANT AHBZ = 35.0        $" Initial Battery AH Capacity "
" AHBR = Battery Ampere-Hour Rating "
CONSTANT AHBR = 35.0
" Cell Open-Circuit Voltages (Volts) "
CONSTANT VOCC1Z = 4.2, VOCC2Z = 4.2, VOCC3Z = 4.2, VOCC4Z = 4.2,...
CONSTANT VOCC5Z = 4.2, VOCC6Z = 4.2, VOCC7Z = 4.2, VOCC8Z = 4.2,...
CONSTANT VOCC9Z = 4.2, VOCC10Z = 4.2
```

```
" CELCON = Cell Controller Logic Macro "
" CELCON Definition "
  MACRO PMACRO CELCON, P
  IF(P(2).GE.P(3))     THEN
      P(1)    =  1.0
  ELSE
      P(1)    =  0.0
  ENDIF
  MACRO END
" CELCAP = Cell Capacitance Selection Logic "
" CELCAP Definition "
  MACRO PMACRO CELCAP, P
  IF(P(3).LE.P(2).AND.P(2).LE.P(4))      THEN
     P(1) = P(5)
  ELSEIF(P(2).LT.P(3).OR.P(2).GT.P(4)) THEN
     P(1) = P(6)
  ENDIF
  MACRO END
" CELCOM = Cell Computation Macro "
" CELCOM Definition "
  MACRO MACRO CELCOM, P
  P(2)    = P(5)-P(6)-P(7)
  P(3)    = (1.0/P(8))*P(2)
  P(1)    = INTEG(P(3), P(9))
  P(4)    = P(10)*P(5)**2 + P(11)*P(6)**2 + P(12)*P(7)**2
  MACRO END
" Thermal Model Parameters "
" MPOLY  = Electrolyte Mass (kg)"
" CPPOLY = Electrolyte Specific Heat (Joules/kg-deg C) "
CONSTANT MPOLY  = 1.9 , CPPOLY  = 1590.0
" MCOP   = Copper Mesh Mass (kg)"
" CPCOP  = Copper Mesh Specific Heat  (Joules/kg-deg C)"
CONSTANT MCOP   = 1.37, CPCOP   = 381.0
" MALUM  = Aluminum Mesh Mass (kg) "
" CPALUM = Aluminum Mesh Specific Heat (Joules/kg-deg C) "
CONSTANT MALUM  = 0.150, CPALUM = 870.0
" MSTL   = Steel Mass (kg) "
" CPSTL  = Steel Specific Heat (Joules/kg-deg C) "
CONSTANT MSTL   = 0.140, CPSTL  = 477.0
" Battery Plastic Can Parameters "
CONSTANT THKB   = 3.0        $" Thickness(mm) "
CONSTANT KSB    = 1.903E-4   $" Conductance Coefficient(Watts/mm-deg C)"
CONSTANT ASB    = 8.48E+5    $" Surface Area (mm**2) "
" MSB14 = Surface Mass (kg) "
CONSTANT  MSB   = 10.0
" CPS   = Surface Specific Heat (Joules/kg-deg C) "
CONSTANT CPSB   = 1590.0
" KOB14 = Convective Heat Transfer Coefficient(Watts/mm**2-Deg C)"
CONSTANT KOB    = 156.45E-6
" Initial Conditions on State Variables "
CONSTANT TIBZ   = 25.0,  TSBZ = 25.0  $" Deg C "
" TOB14 = Outside Air Temperature "
CONSTANT TOB    = 25.0                $" Deg C "
             " Initial Region Computations "
INITIAL
" General Computed Parameters "
PI      = ACOS(-1.0)
" Generator Computed Parameters "
TSIDLE  = TSENG  + TRIDLE
TAENG   = TSIDLE + TIDLE
```

```
TCENG    = TAENG   + TRENG
TDENG    = TCENG   + TCRUISE
TSTOP    = TDENG   + TFENG
" Thermal model Computed Parameters "
" CTHIB = Battery Interior Thermal Capacitance (Watt-Sec/deg C) "
CTHIB    = MPOLY*CPPOLY+MCOP*CPCOP+MALUM*CPALUM+MSTL*CPSTL
" RSB    = Surface Conductive Heat Transfer Resistance(deg C/Watts)"
RSB      = THKB/(KSB*ASB)
" CTHSB = Battery Surface Thermal Capacitance (Watt-Sec/deg C) "
CTHSB    = MSB*CPSB
" ROB    = Surface Convective Heat Transfer Coefficient(deg C/Watts) "
ROB      = 1.0/(KOB*ASB)
IAGENS   = 0.0
IBGENS   = 0.0
ICGENS   = 0.0
END INITIAL
                    " Dynamic and Derivative Region Computations "
DYNAMIC
 DERIVATIVE EQS
                    " Simulation Controls "
 ALGORITHM ISTART = 5, IRUN = 5
 CINTERVAL CI     = 25.0E-6
 NSTEPS    NST    = 25
 MINTERVAL HMINT = 1.0E-20
 " Engine Speed Command Model "
 PROCEDURAL (NENGS = T)
   IF(T.LT.TSENG)                              THEN
     NENGS    = 0.0
   ELSEIF(TSENG.LE.T.AND.T.LT.TSIDLE)          THEN
     NENGS    = NVIDLE
   ELSEIF(TSIDLE.LE.T.AND.T.LT.TAENG)          THEN
     NENGS    =  0.0
   ELSEIF(TAENG.LE.T.AND.T.LT.TCENG)           THEN
     NENGS    = NVRENG
   ELSEIF(TCENG.LE.T.AND.T.LT.TDENG)           THEN
     NENGS    =  0.0
   ELSEIF(TDENG.LE.T.AND.T.LT.TSTOP)           THEN
     NENGS    = -NVRENG
   ELSEIF(T.GE.TSTOP)                          THEN
     NENGS    =  0.0
   ENDIF
 END
 NENG  = INTEG(NENGS,0.0)
 " NGEN     = Generator Speed (rpm) "
 " MODTEST = 1.0 Generator Test is On "
 " MODTEST = 0.0 Generator Test is Off "
 PROCEDURAL(NGEN = MODTEST, NENG, NEZ)
   IF(MODTEST.LT.0.5)       THEN
      NGEN    = GRGEN*NENG
   ELSEIF(MODTEST.GE.0.5)   THEN
      NGEN    = GRGEN*NEZ
   ENDIF
 END
 " Voltage regulator Model "
 VGBFBK = VGB42
 VGBERR = VGBREF-VGBFBK
 VGBCON = GPLEC*VGBERR + GILEC*INTEG(VGBERR, VGPCONZ)
 PROCEDURAL(DREG = VGBCON)
   IF(VGBCON.GE.VLECPL)                          THEN
      DREG  = DREGMX
```

```
  ELSEIF( 0.0.LE.VGBCON.AND.VGBCON.LT.VLECPL) THEN
     DREG   = (DREGMX/VLECPL)*VGBCON
  ELSE
     DREG   = 0.0
  ENDIF
END
VFCON   = DREG*VGBFBK
LFGEN   = LFGENT(IFGEN)
RFGEN   = RFGENO*(1.0+ALPHAW*(TIG-TIGREF))
" IFGENS = Field Current Derivative (amp/sec)"
IFGENS  = (1.0/LFGEN)*(VFCON-RFGEN*IFGEN-LAMFSC)*MODCON
IFGEN   = INTEG(IFGENS,IFGENZ)
" LEC Controller Model "
PROCEDURAL(VLEC = VGBCON)
  IF(VGBCON.LE.VLECNL)        THEN
     VLEC = 1.0
  ELSE
     VLEC = 0.0
  ENDIF
END
" Generator Model "
" OMGGEN  = Generator Electric Frequency (rad/sec) "
OMGGEN = NGEN*(2.0*PI/60.0)*NPPGEN
" Computation of Circuit Parameters "
" Stator Phase Resistances (Ohms) "
RAGEN   = RAGENO*(1.0+ALPHAW*(TIG-TIGREF))
RBGEN   = RBGENO*(1.0+ALPHAW*(TIG-TIGREF))
RCGEN   = RCGENO*(1.0+ALPHAW*(TIG-TIGREF))
" Stator-Field Mutual Inductances (Henry) "
LMFA    = LMFT(IFGEN,NGEN)
LMFB    = LMFA
LMFC    = LMFA
" LAMF  = Field flux induced by phase currents (Weber) "
" LAMFS = Field Voltage induced by phase currents (Volts)"
" LAMFSC= Computed LAMFS "
LAMF    = LMFA*COS(OMGGEN*T)*IAGEN+...
          LMFB*COS(OMGGEN*T-(2.0*PI/3.0))*IBGEN+...
          LMFC*COS(OMGGEN*T+(2.0*PI/3.0))*ICGEN
LAMFSC  = (1.0/TAUDIF)*(LAMF-LAMFC)
LAMFC   = INTEG(LAMFSC, 0.0)
" Compute Synchronous Inductances (Henry) "
LSA     = LST(NGEN)
LSB     = LSA
LSC     = LSA
" Back emf Voltages "
" VAGEN = Field-Phase A Back EMF Voltage (Volts) "
VMFA    = LMFA*COS(OMGGEN*T)*IFGENS
VAGEN   = OMGGEN*LMFA*SIN(OMGGEN*T)*IFGEN-VMFA
" VBGEN = Field-Phase B Back EMF Voltage (Volts) "
VMFB    = LMFB*COS(OMGGEN*T-(2.0*PI/3.0))*IFGENS
VBGEN   = OMGGEN*LMFB*SIN(OMGGEN*T-(2.0*PI/3.0))*IFGEN-VMFB
" VCGEN = Field-Phase C Back EMF Voltage (Volts) "
VMFC    = LMFC*COS(OMGGEN*T+(2.0*PI/3.0))*IFGENS
VCGEN   = OMGGEN*LMFC*SIN(OMGGEN*T+(2.0*PI/3.0))*IFGEN-VMFC
" VGENPK = Peak back emf Voltage (Volts) "
" VGENXR = Phase Voltage Crossing Level (Volts) "
VGENPK = OMGGEN*LMFA*IFGEN
VGENXR = 0.5*VGENPK
" VSUMGEN = Generator Voltages Sum "
VSUMGEN= VAGEN + VBGEN + VCGEN
```

```
" VGB42 = Generator 42 Volt Bus Voltage (Volts) "
VGB42    = (1.0-SWLD42)*VB10+VDT(ISGEN)-SWLD42*VDT(-ISGEN)+...
           (1.0-SWLD42)*VDT(ISGEN)
" Computation of Delta Winding Line-to-Line Voltages "
" VABGEN Computation "
PROCEDURAL(VABGEN = VAGEN,VBGEN,VCGEN,VGB42,VGENPK,VGENXR)
 IF(VGENPK.LE.VGB42.AND.VGENXR.LE.VGB42)                              THEN
      VABGEN    = VAGEN
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.LE.VGB42.AND.VAGEN.GE.0.0...
                                                .AND.VCGEN.LE.0.0)    THEN
      VABGEN    = VGB42
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.GT.VGB42.AND.VBGEN.GT.VCGEN...
                                                .AND.VAGEN.GT.VCGEN)  THEN
      VABGEN    = VGB42
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.LE.VGB42.AND.VAGEN.LE.0.0...
                                                .AND.VCGEN.LE.0.0)    THEN
      VABGEN    = 0.0
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.GT.VGB42.AND.VAGEN.LE.VCGEN...
                                                .AND.VCGEN.LE.VBGEN)  THEN
      VABGEN    = 0.0
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.LE.VGB42.AND.VAGEN.LE.0.0...
                                                .AND.VCGEN.GE.0 0)    THEN
      VABGEN    = -VGB42
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.GT.VGB42.AND.VBGEN.LT.VCGEN...
                                                .AND.VAGEN.LT.VCGEN)  THEN
      VABGEN    = -VGB42
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.LE.VGB42.AND.VAGEN.GE.0.0...
                                                .AND.VCGEN.GE.0.0)    THEN
      VABGEN    = 0.0
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.GT.VGB42.AND.VAGEN.GE.VCGEN...
                                                .AND.VCGEN.GE.VBGEN)  THEN
      VABGEN    = 0.0
 ENDIF
END
" VBCGEN Computation "
PROCEDURAL(VBCGEN = VAGEN,VBGEN,VCGEN,VGB42,VGENPK,VGENXR)
 IF(VGENPK.LE.VGB42.AND.VGENXR.LE.VGB42)                              THEN
      VBCGEN    = VBGEN
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.LE.VGB42.AND.VBGEN.GE.0.0...
                                                .AND.VAGEN.LE.0.0)    THEN
      VBCGEN    = VGB42
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.GT.VGB42.AND.VCGEN.GT.VAGEN...
                                                .AND.VBGEN.GT.VAGEN)  THEN
      VBCGEN    = VGB42
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.LE.VGB42.AND.VBGEN.LE.0.0...
                                                .AND.VAGEN.LE.0.0)    THEN
      VBCGEN    = 0.0
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.GT.VGB42.AND.VBGEN.LE.VAGEN...
                                                .AND.VAGEN.LE.VCGEN)  THEN
      VBCGEN    = 0.0
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.LE.VGB42.AND.VBGEN.LE.0.0...
                                                .AND.VAGEN.GE.0.0)    THEN
      VBCGEN    = -VGB42
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.GT.VGB42.AND.VCGEN.LT.VAGEN...
                                                .AND.VBGEN.LT.VAGEN)  THEN
      VBCGEN    = -VGB42
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.LE.VGB42.AND.VBGEN.GE.0.0...
                                                .AND.VAGEN.GE.0.0)    THEN
      VBCGEN    = 0.0
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.GT.VGB42.AND.VCGEN.LE.VAGEN...
```

```
                                                    .AND.VAGEN.LE.VBGEN)    THEN
      VBCGEN    = 0.0
 ENDIF
END
" VCAGEN Computation "
PROCEDURAL(VCAGEN = VAGEN,VBGEN,VCGEN,VGB42,VGENPK,VGENXR)
 IF(VGENPK.LE.VGB42.AND.VGENXR.LE.VGB42)                              THEN
      VCAGEN    = VCGEN
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.LE.VGB42.AND.VCGEN.GE.0.0...
                                                    .AND.VBGEN.LE.0.0)    THEN
      VCAGEN    = VGB42
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.GT.VGB42.AND.VAGEN.GT.VBGEN...
                                                    .AND.VCGEN.GT.VBGEN)  THEN
      VCAGEN    = VGB42
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.LE.VGB42.AND.VCGEN.LE.0.0...
                                                    .AND.VBGEN.LE.0.0)    THEN
      VCAGEN    = 0.0
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.GT.VGB42.AND.VCGEN.LE.VBGEN...
                                                    .AND.VBGEN.LE.VAGEN)  THEN
      VCAGEN    = 0.0
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.LE.VGB42.AND.VCGEN.LE.0.0...
                                                    .AND.VBGEN.GE.0.0)    THEN
      VCAGEN    = -VGB42
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.GT.VGB42.AND.VAGEN.LT.VBGEN...
                                                    .AND.VCGEN.LT.VBGEN)  THEN
      VCAGEN    = -VGB42
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.LE.VGB42.AND.VCGEN.GE.0.0...
                                                    .AND.VBGEN.GE.0.0)    THEN
      VCAGEN    = 0.0
 ELSEIF(VGENPK.GT.VGB42.AND.VGENXR.GT.VGB42.AND.VAGEN.LE.VBGEN...
                                                    .AND.VBGEN.LE.VCGEN)  THEN
      VCAGEN    = 0.0
 ENDIF
END
VLSUMGEN= VABGEN+VBCGEN+VCAGEN
" Computation of Generator Stator Phase Currents (Amps) "
" IAGEN, IBGEN, ICGEN Computation "
IAGENS   = (1.0/LSA)*(VAGEN-VABGEN-RAGEN*IAGEN)
IAGEN    = INTEG(IAGENS, 0.0)
IBGENS   = (1.0/LSB)*(VBGEN-VBCGEN-RBGEN*IBGEN)
IBGEN    = INTEG(IBGENS, 0.0)
ICGENS   = (1.0/LSC)*(VCGEN-VCAGEN-RCGEN*ICGEN)
ICGEN    = INTEG(ICGENS, 0.0)
" ISUMGEN= Sum of Generator Phase Currents "
ISUMGEN  = IAGEN+IBGEN +ICGEN
" Compute Line Currents IALGEN, IBLGEN, ICLGEN "
IALGEN   = IAGEN - ICGEN
IBLGEN   = IBGEN - IAGEN
ICLGEN   = ICGEN - IBGEN
" ILSUMGEN = Sum of Generator Line Currents '
ILSUMGEN = IALGEN+IBLGEN+ICLGEN
" Compute Generator Rectified DC Current ISGEN "
PROCEDURAL (IADCGEN = IALGEN)
  IF (IALGEN.GE.0.0)  THEN
     IADCGEN = IALGEN
  ELSE
     IADCGEN = 0.0
  ENDIF
END
PROCEDURAL (IBDCGEN = IBLGEN)
```

```
  IF (IBLGEN.GE.0.0)  THEN
     IBDCGEN = IBLGEN
  ELSE
     IBDCGEN = 0.0
  ENDIF
END
PROCEDURAL (ICDCGEN = ICLGEN)
  IF (ICLGEN.GE.0.0)  THEN
     ICDCGEN = ICLGEN
  ELSE
     ICDCGEN = 0.0
  ENDIF
END
ISGEN  = IADCGEN + IBDCGEN + ICDCGEN
" Computation of Generator Torque (Nm) "
" PWGEN = Generator Power Output (Watts) "
" TQGEN = Generator Electrical Torque (Nm) "
PWGEN   = (VAGEN*IAGEN+VBGEN*IBGEN+VCGEN*ICGEN)
TQGEN   = PWGEN/(NGEN*(2.0*PI/60.0)+1.0)
" PWLSGEN = Stator Power Loss (Watts) "
PWLSGEN = RAGEN*IAGEN**2+RBGEN*IBGEN**2+RCGEN*ICGEN**2
" PWLFGEN = Field Power Loss (Watts) "
PWLFGEN  = RFGEN*IFGEN**2
" PWLRGEN = Rectifier Power Loss (Watts) "
PWLDA  = VDT(IADCGEN)*IADCGEN
PWLDB  = VDT(IBDCGEN)*IBDCGEN
PWLDC  = VDT(ICDCGEN)*ICDCGEN
PWLRGEN= PWLDA+PWLDB+PWLDC
" PWBUS = Generator Power Output at the Bus (Watts)"
PWBUS  = VGB42*ISGEN
" Generator Thermal Model "
 TIG    = 25.0
 " 42 Volt Loads "
 " ISM  = Starter Motor Load Current (Amps) "
 ISM    = ISMAMP*( STEP(TSMON,T)-STEP(TSMOFF,T))
 " PWLSM = Starter Motor Power Loss (Watts) "
 PWLSM  = RSM*ISM**2
 " PWSM  = Starter Motor Power Output (Watts) "
 " TSM   = Motor Torque (Nm) "
 TSM    = KTSM*ISM
 PWSM   = TSM*(NEZ*(2.0*PI/60.0))
 " IPSM = Power Steering Motor Load Current (Amps) "
 PROCEDURAL(IPSM = T)
   IF(T.LT.TPSMON)                          THEN
      IPSM   =  0.0
   ELSEIF(TPSMON.LE.T.AND.T.LT.TPSMOFF)   THEN
      IPSM   = IPSMAMP*( STEP(TPSMON,T)- STEP(TPSMOFF,T))
   ELSE
      IPSM   =  0.0
   ENDIF
 END
 " PWLPSM = Power Steering Motor Power Loss (Watts) "
 PWLPSM  = RPSM*IPSM**2
 " PWPSM = Power Steering Motor Power Output (Watts) "
 " TPSM  = Motor Torque (Nm) "
 TPSM    = KTPSM*IPSM
 PWPSM   = TPSM*OMGPS
 " LiPo Battery Model "
 " State Equations for the electric analog circuit model "
 " Positive Current = Charge, Negative Current = Discharge "
```

```
" CFRC = Resistance Correction Factor "
CFRC    = CFRBT(TIB)
" Battery Discharge/Charge Current Computation "
IBC10   = (1.0-SWLD42)*(ISGEN-ISM-IPSM)
" Computation of Battery Storage Capacitances "
CSB     = CSBT(IBC10)
CSBL    = CSBLT(IBC10)
" Cell-10 "
SWC10   = CELCON(VC10,VCON10)
VOCC10,ICC10,VOCC10S,PWLC10 = CELCOM(IBC10,ILC10,ICON10,CSC10,...
                                     VOCC10Z,RIC10,RLC10,RCON10)
CSC10   = CELCAP(VC10,VCMN,VCMX,CSB,CSBL)
RIC10   = CFRC*RIC10I
RLC10   = CFRC*RLC10I
ILC10   = (VOCC10/RLC10)
RCON10  = CFRC*RCON10I
ICON10  = (SWC10*VOCC10)/RCON10
VC10    = VOCC10+ RIC10*IBC10
VB10    = VC10+VB9
" Cell-9 "
IBC9    = ICC10
SWC9    = CELCON(VC9,VCON9)
VOCC9,ICC9,VOCC9S,PWLC9 = CELCOM(IBC9,ILC9,ICON9,CSC9,...
                                 VOCC9Z,RIC9,RLC9,RCON9)
CSC9    = CELCAP(VC9,VCMN,VCMX,CSB,CSBL)
RIC9    = CFRC*RIC9I
RLC9    = CFRC*RLC9I
ILC9    = (VOCC9/RLC9)
RCON9   = CFRC*RCON9I
ICON9   = (SWC9*VOCC9)/RCON9
VC9     = VOCC9+ RIC9*IBC9
VB9     = VC9+VB8
" Cell-8 "
IBC8    = ICC9
SWC8    = CELCON(VC8,VCON8)
VOCC8,ICC8,VOCC8S,PWLC8 = CELCOM(IBC8,ILC8,ICON8,CSC8,...
                                 VOCC8Z,RIC8,RLC8,RCON8)
CSC8    = CELCAP(VC8,VCMN,VCMX,CSB,CSBL)
RIC8    = CFRC*RIC8I
RLC8    = CFRC*RLC8I
ILC8    = (VOCC8/RLC8)
RCON8   = CFRC*RCON8I
ICON8   = (SWC8*VOCC8)/RCON8
VC8     = VOCC8+ RIC8*IBC8
VB8     = VC8+VB7
" Cell-7 "
IBC7    = ICC8
SWC7    = CELCON(VC7,VCON7)
VOCC7,ICC7,VOCC7S,PWLC7 = CELCOM(IBC7,ILC7,ICON7,CSC7,...
                                 VOCC7Z,RIC7,RLC7,RCON7)
CSC7    = CELCAP(VC7,VCMN,VCMX,CSB,CSBL)
RIC7    = CFRC*RIC7I
RLC7    = CFRC*RLC7I
ILC7    = (VOCC7/RLC7)
RCON7   = CFRC*RCON7I
ICON7   = (SWC7*VOCC7)/RCON7
VC7     = VOCC7+RIC7*IBC7
VB7     = VC7+VB6
" Cell-6 "
IBC6    = ICC7
```

```
SWC6    = CELCON(VC6,VCON6)
VOCC6,ICC6,VOCC6S,PWLC6 = CELCOM(IBC6,ILC6,ICON6,CSC7,...
                                 VOCC6Z,RIC6,RLC6,RCON6)
CSC6    = CELCAP(VC6,VCMN,VCMX,CSB,CSBL)
RIC6    = CFRC*RIC6I
RLC6    = CFRC*RLC6I
ILC6    = (VOCC6/RLC6)
RCON6   = CFRC*RCON6I
ICON6   = (SWC6*VOCC6)/RCON6
VC6     = VOCC6+ RIC6*IBC6
VB6     = VC6+ VB5
" Cell-5 "
IBC5    = ICC6
SWC5    = CELCON(VC5,VCON5)
VOCC5,ICC5,VOCC5S,PWLC5 = CELCOM(IBC5,ILC5,ICON5,CSC5,...
                                 VOCC5Z,RIC5,RLC5,RCON5)
CSC5    = CELCAP(VC5,VCMN,VCMX,CSB,CSBL)
RIC5    = CFRC*RIC5I
RLC5    = CFRC*RLC5I
ILC5    = (VOCC5/RLC5)
RCON5   = CFRC*RCON5I
ICON5   = (SWC5*VOCC5)/RCON5
VC5     = VOCC5+RIC5*IBC5
VB5     = VC5+ VB4
" Cell-4 "
IBC4    = ICC5
SWC4    = CELCON(VC4,VCON4)
VOCC4,ICC4,VOCC4S,PWLC4 = CELCOM(IBC4,ILC4,ICON4,CSC4,...
                                 VOCC4Z,RIC4,RLC4,RCON4)
CSC4    = CELCAP(VC4,VCMN,VCMX,CSB,CSBL)
RIC4    = CFRC*RIC4I
RLC4    = CFRC*RLC4I
ILC4    = (VOCC4/RLC4)
RCON4   = CFRC*RCON4I
ICON4   = (SWC4*VOCC4)/RCON4
VC4     = VOCC4+RIC4*IBC4
VB4     = VC4+ VB3
" Cell-3 "
IBC3    = ICC4
SWC3    = CELCON(VC3,VCON3)
VOCC3,ICC3,VOCC3S,PWLC3 = CELCOM(IBC3,ILC3,ICON3,CSC3,...
                                 VOCC3Z,RIC3,RLC3,RCON3)
CSC3    = CELCAP(VC3,VCMN,VCMX,CSB,CSBL)
RIC3    = CFRC*RIC3I
RLC3    = CFRC*RLC3I
ILC3    = (VOCC3/RLC3)
RCON3   = CFRC*RCON3I
ICON3   = (SWC3*VOCC3)/RCON3
VC3     = VOCC3 + RIC3*IBC3
VB3     = VC3 + VB2
" Cell-2 State Equations "
IBC2    = ICC3
SWC2    = CELCON(VC2,VCON2)
VOCC2,ICC2,VOCC2S,PWLC2 = CELCOM(IBC2,ILC2,ICON2,CSC2,...
                                 VOCC2Z,RIC2,RLC2,RCON2)
CSC2    = CELCAP(VC2,VCMN,VCMX,CSB,CSBL)
RIC2    = CFRC*RIC2I
RLC2    = CFRC*RLC2I
ILC2    = (VOCC2/RLC2)
RCON2   = CFRC*RCON2I
```

```
  ICON2    = (SWC2*VOCC2)/RCON2
  VC2      = VOCC2+ RIC2*IBC2
  VB2      = VC2+VB1
  " Cell-1 State Equations "
  IBC1     = ICC2
  SWC1     = CELCON(VC1,VCON1)
  VOCC1,ICC1,VOCC1S,PWLC1 = CELCOM(IBC1,ILC1,ICON1,CSC1,...
                                   VOCC1Z,RIC1,RLC1,RCON1)
  CSC1     = CELCAP(VC1,VCMN,VCMX,CSB,CSBL)
  RIC1     = CFRC*RIC1I
  RLC1     = CFRC*RLC1I
  ILC1     = (VOCC1/RLC1)
  RCON1    = CFRC*RCON1I
  ICON1    = (SWC1*VOCC1)/RCON1
  VC1      = VOCC1+ RIC1*IBC1
  VB1 = VC1
  " AHB    = Net Battery Ampere-Hour Capacity (AH) "
  AHBS     = (1.0/3600.0)*ICC10
  AHB      = INTEG(AHBS,AHBZ)
  " State Of Charge (SOC) "
  SOC    = (AHB/AHBR)
  " Power Computations "
  " PWBC  = Power at the Battery Output (Watts) "
  PWBC     = VB10*IBC10
  " State Equations for the Thermal Model "
  " TIBS  = Rate of Interior Battery Temperature (Deg C/sec)"
  " TIB   = Interior Battery Temperature (Deg C)"
  " CTHIB = Battery Interior Capacitance (Watt-Sec / Deg C) "
  " HINB  = Input Heating Power (Watts) "
  HINB = PWLC1+PWLC2+PWLC3+PWLC4+PWLC5+PWLC6+PWLC7+PWLC8+PWLC9+PWLC10
  " HSB   = Interor-Surface Conduction Heat Transfer (Watts) "
  " TSB   = Battery Surface Temperature (deg C) "
  " RSB   = Interior-Surface Conduction H-T Coefficient(deg C/Watts)"
  HSB      = (TIB-TSB)/RSB
  TIBS     = (1.0/CTHIB)*(HINB-HSB)
  TIB      = INTEG(TIBS,TIBZ)
  " TSBS  = Rate of Battery Surface Temperature (deg C / sec) "
  " TSB   = Battery Surface Temperature (deg C) "
  " CTHS  = Battery Surface Thermal Capacitance (Watt-Sec/deg C ) "
  " HOB   = Surface-to-Outside Convective Heat Transfer (Watts) "
  " TOB   = Outside Air Temperature (deg C ) "
  " ROB   = Surface-to-Outside Convective H-T Coefficient(deg C/Watts)"
  HOB      = (TSB-TOB)/ROB
  TSBS     = (1.0/CTHSB)*(HSB-HOB)
  TSB      = INTEG(TSBS,TSBZ)
   END DERIVATIVE
 TERMT(T.GE.TFIN)
 END DYNAMIC
                  " Terminal Region Computations "
 TERMINAL
 END TERMINAL

END PROGRAM
```